

# UserModel +userID: int +name: string +password: string +weight: float +height: float +calfCir: float +hipCir: float +hipCir: float +age: string +chestCir: float +upperArmCir: float

### FeedbackModel +workoutIntensitiy: int +workoutDuration: int +exerciseVariaty: int

## WorkoutnoutineModels +workoutroutineID +date : string +exerciseIDs : vector<int> +targetReps : vector<int> +targetSets : vector<int> +targetWeight : vector<int>

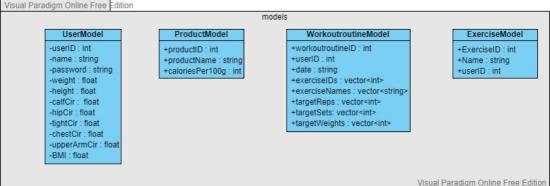
```
ProductModel
+productID: int
+productName: string
+ caloriesPer100g: int
```

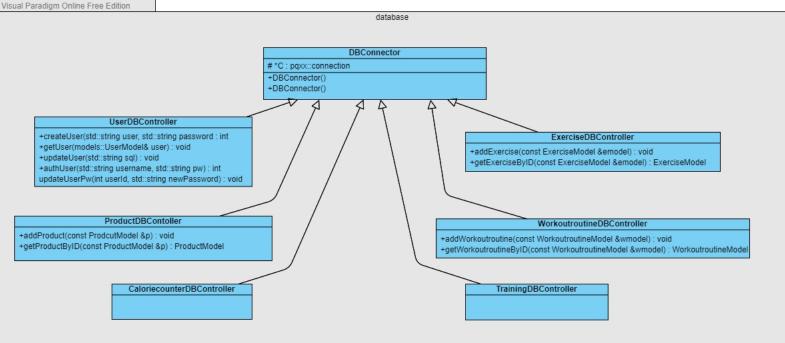
ExerciseModel
+exerciseID : int
+name : string
+userID

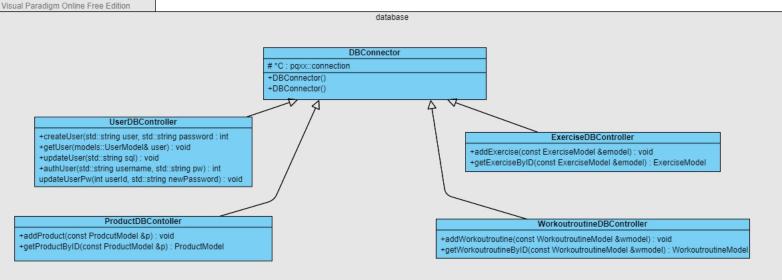
## WorkoutModel

+endtime : int

Visual Paradigm Online Free Edition







request

### APIBasicController

- +handleGET(http\_request message): void
- +handlePUT(http\_request message, json::value body): void
- +handlePOST(http\_request message, json::value body): void +handleDELTE(http\_request message): void

### UserRequestController

- +handleGET(http\_request message): void
- +handlePUT(http\_request\_message, json::value body); void
- +handlePOST(http\_request message, json::value body): void +handleDELTE(http\_request message): void
- +UserRequestController(const http://request.&message, const method.&method)

### WorkoutroutineRequestController

- +handleGET(http\_request\_message); void
- +handlePUT(http\_request message, json::value body): void
- +handlePOST(http\_request message, json::value body): void
- +handleDELTE(http\_request message): void
- +WorkoutroutineRequestController(const http\_request &message, const method &method)

### FeedbackRequestController

- +handleGET(http\_request message): void
- +handlePUT(http\_request message, json::value body): void
- +handlePOST(http\_request message, json::value body); void
- +handleDELTE(http\_request message): void
- +FeedbackRequestController(const http\_request &message, const method &method)

### AuthenticationRequestController

- +handleGET(http\_request\_message); void
- +handlePUT(http\_request message, json::value body): void
- +handlePOST(http\_request message, json::value body): void
- +handleDELTE(http\_request message): void
- +AuthenticationRequestController(const http\_request &message, const method &method)

### ExerciseRequestController

- +handleGET(http\_request message): void
- +handlePUT(http\_request message, json::value body): void +handlePOST(http\_request message, json::value body): void
- +handleDELTE(http\_request message): void
- +ExerciseRequestController(const http://request.&message, const method.&method)

### ProductRequestController

- +handleGET(http\_request\_message); void
- +handlePUT(http\_request message, json::value body): void +handlePOST(http\_request message, json::value body): void
- +handleDELTE(http\_request message): void
- +ProductRequestController(const http\_request &message, const method &method)

### CaloriecounterRequestController

- +handleGET(http\_request message): void
- +handlePUT(http\_request message, json::value body): void
- +handlePOST(http\_request\_message, json::value body); void
- +handleDELTE(http\_request message): void
- +CaloriecounterRequestController(const http\_request &message, const method &method)

### TrainingRequestController

- +handleGET(http\_request message): void
- +handlePUT(http\_request message, json::value body): void
- +handlePOST(http\_request\_message, json::value\_body); void
- +handleDELTE(http\_request message): void
- +TrainingRequestController(const http://request.kmessage.const/method)